



## Intermediate Java

**The Intermediate Java course** examines common language features and APIs required to develop complex stand-alone Java applications. The course builds on the Introduction to Java course and examines topics such as lambda expressions, generic programming, and new features in Java 8.

### Productivity Objectives:

#### This Class Does Include:

- Key Design Patterns
- Using and Defining Generic Classes and Methods
- Introduction to Functional Programming with Higher Order Functions and Immutable Data
- Java 8 Streams
- Dynamic Java

#### This Class Does Not Include:

- Basic OO concepts; Encapsulation, Inheritance, Generalization, Instance vs Static Features
- Java syntax from Java 7 other than Generics

**Course Duration:** This course will be delivered in 3 Days

### Course Outline:

#### Design Patterns

- Factory
- Singleton
- Builder
- Command

#### Lambda Expressions

- Lambda Syntax
- Constraints on Using Lambdas
- Functional Interfaces
- Method References

## **Functional Programming Concepts**

- Higher Order Functions
- Working with Immutable Data
- Closures

## **Streams**

- Terminal vs non-terminal operations
- forEach, filter, map, flatMap
- Reduction - reduce and collect
- Collectors
- Primitive Stream Types
- Parallel Streams
- Optional

## **Effective Exception Handling**

- Review of Basic Exception Handling concepts
- Resource Handling with Exceptions
- Exceptions and Lambdas
- Going Exceptionless with Try Monads

## **Generics**

- Creating Generic Types
- Generic Methods
- Type Variable Bounds
- Type Erasure
- Inheritance and Generics
- Co- and Contra-Variance

## **Reflection and Annotations**

- Examining a Loaded Class
- Loading a Class Dynamically
- Finding Annotations on Syntactic Elements of Loaded Classes
- Interacting with Methods and Fields Dynamically
- Defining Annotations
- Controlling Applicability and Retention of Annotations
- Using Annotations with Key-Value Pairs